MLP Coursework 4 - G1

s1756643, s1757323, s1777706

Abstract

We investigate the use of Mask R-CNN for the task of nuclei segmentation in biomedical images. We explore the use of different hyper parameters settings, sensible for the task of nuclei segmentation.

By adjusting the non-suppression threshold, number of RoI, anchor and and image size, we find that the Mask R-CNN model outperforms our baseline U-Net model.

Future work should explore more hyper parameter tuning of the Mask R-CNN model for nuclei segmentation tasks with several different nuclei classes. Indeed, we show that this improvement comes at the price of a more complex model which requires more time to be trained and a deeper understanding to be used effectively.

1. Introduction

The currently ongoing Kaggle competition 'Spot the Nuclei' addresses the need for detection and segmentation of nuclei in medical images in order to accelerate medical research f. In order to analyse the effect of drugs or medication, it is necessary to locate the cell nuclei and observe its reaction to it. Currently, this typically involves a trained professional manually analysing every single image, which is time and cost intensive.

One of the major challenges working with medical images is the sparsity of available data for this task. Due to the immense work involved in annotating biomedical images, it is important to find a model that brings results for little available data. Furthermore, the variability of biomedical images due to different staining techniques and dye concentration asks for a model that generalises well for a range of slide preparations (Irshad et al., 2014).

In coursework 3, we implemented U-Net, a fully convolutional network architecture, as our baseline. We showed that the use of pre-processed images and data augmentation improves a models performance and generalisation error. (Aquilina et al., 2018)

The advances in machine learning give hope that these problems may be solvable by the use of state of the art instance segmentation techniques. Mask R-CNN(He et al., 2017) is a promising model recently introduced by Facebook AI Research. It builds upon previous successes of the Fast R-CNN model (Girshick, 2015), the Faster R-CNN model (Ren et al., 2015) as well as the Fully Convolutional Network (FCN) (Long et al., 2015). He et al. (2017) introduces the addition of a segmentation mask head as an extension of Faster RCNN. Mask R-CNN has been shown to be an effective tool when it comes to instance segmentation (He et al., 2017). The improvements in performance and its ability to do inference at a quick rate enables its application to areas such as the recognition of cars and pedestrians in self-driving cars (He et al., 2017). These advances lead us to our first research question: *Can its potential be translated to the task of nuclei segmentation in medical images?*

However, Mask R-CNN introduces a much more complex model which is not as quick and easy to train as our baseline U-Net model and contains many more hyperparameters (63,733,406 vs 1,941,105 parameters). Which introduces our second research question: *Is using Mask R-CNN a feasible choice for the nuclei segmentation task?*

Our main objective is to adapt this model in order generalize in the task of image segmentation for microscopic images. With this coursework, we aim to explore the possibilities of this model in the domain of medical research and validates whether Mask R-CNN is an effective tool for biomedical image segmentation.

This will be done by conducting several experiments in order to assess the performance of Mask R-CNNs for the nuclei segmentation task. We explore whether choosing sensible hyper parameter settings allows the Mask R-CNN model to perform nuclei segmentation more effectively than the previously explored baseline U-Net model.

The outline of this report is as follows:

- First, we present the architecture of Mask R-CNN, the pre-processing steps that we used and data augmentation techniques (Section 2).
- In Section 3, we run several experiments to tune the Mask R-CNN specific hyperparameters to achieve best performance on the nuclei segmentation task. All findings are compared to our U-Net baseline model, presented in coursework 3.
- In Section 4, we relate our findings to the current research in the field of medical object recognition and segmentation.
- Finally, we discuss and suggest future work in the field.

2. Methodology

In coursework 3, we reviewed the architectures of R-CNN, Fast R-CNN, Faster R-CNN as well as briefly introduced Mask R-CNN, a recently proposed model used for instance segmentation (Aquilina et al., 2018; Girshick et al., 2014; Girshick, 2015; Ren et al., 2015; He et al., 2017). In this section we will delve deeper into the details of the implementation we used[1]. First, let us summarise the model by its high level components.

Mask R-CNN consists of three main component networks:

- Backbone Network
- Region Proposal Network (RPN)
- Network Heads
  - Bounding Box Classifier Head
  - Bounding Box Regressor Head
  - Segmentation Mask Head

The latter of which contains 3 different networks for the independent classification, regression and segmentation of the image. Let us deconstruct this model by considering each of these components individually.

2.1. Backbone Network

The backbone network consists of a convolutional neural network (CNN) with the sole purpose of feature extraction. The resultant feature maps generated by this network are used by the RPN to propose regions of interest. He et al. (2017) explore various backbone architectures including ResNet-50, ResNet-101 and ResNeXt-101(He et al., 2017; 2016; Xie et al., 2017). Of the three models, ResNet-101 was chosen for experimentation. ResNet-101 consists of five layers with homogeneous building blocks similar to that in Figure 1. The model structure for ResNet is summarised in Table 1. This implementation uses a ReLU layer wherever an activation layer is required (Nair & Hinton, 2010).

He et al. (2017) also implemented a Feature Pyramid Network (FPN) based on each of the previously mentioned backbones (Lin et al., 2017). This uses the pyramid structure typically seen in CNNs and duplicates it as seen in Figure 2. The main aim of the FPN is to allow building high level semantic feature maps at varying scales. Similar to (He et al., 2017), we use a FPN based on ResNet-101 in our implementation.

2.2. Region Proposal Network

The purpose of the RPN is to take an image of any dimension and output bounding box proposals for objects. This lightweight network consists of a square window which is scanned across the feature maps generated by the backbone. This $n \times n$ convolutional layer is fed into two fully convolutional layers, one for box regression and one for classification. The positions to which the $n \times n$ network

[1]Code adapted from: https://github.com/matterport/Mask_RCNN

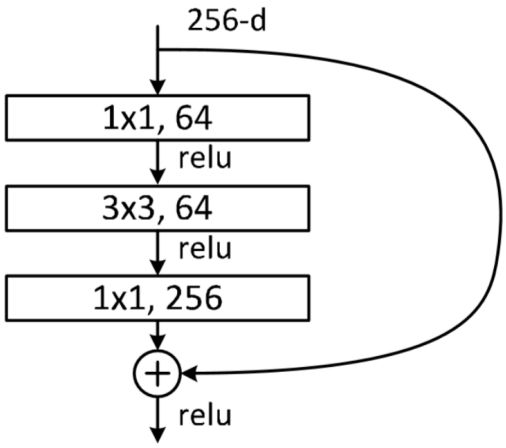

Figure 1. ResNet C2 Layer Building Block (He et al., 2016)

Table 1. ResNet-101 Model Structure

Layer Name	Building Block
C1	7×7, 64, Stride 2
C2	3×3 Max Pool, Stride 2 $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
C3	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$
C4	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$
C5	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$

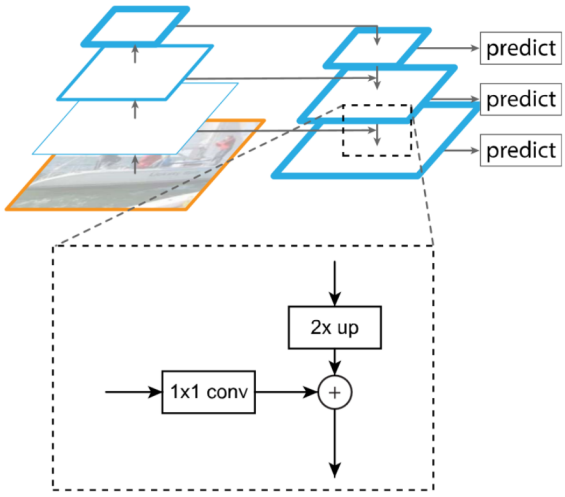

Figure 2. Feature Pyramid Network (FPN) (Lin et al., 2017)

is moved are called anchors which can vary in separation (stride). At each anchor, a set of different sized boxes (anchor scales) of varying aspect ratio are proposed. Boxes which contain a significant portion of the underlying target mask are refined such that they encapsulate the target object better (Ren et al., 2015). In our case, the RPN has not one but multiple feature maps to propose bounding boxes on (due to the FPN). To do so, it has a fixed set of anchor scales, one corresponding to each feature map in the pyramid. It therefore selects a pyramid feature map to be used, based on the scale of the box.

As shown in Figure 3, the outputs of this network are the following:

- Background or Foreground classification
  - $2 \times k$ class softmax
- Refined Bounding Box
  - $4 \times k$ coordinates

where k is the number of anchor boxes at each anchor.

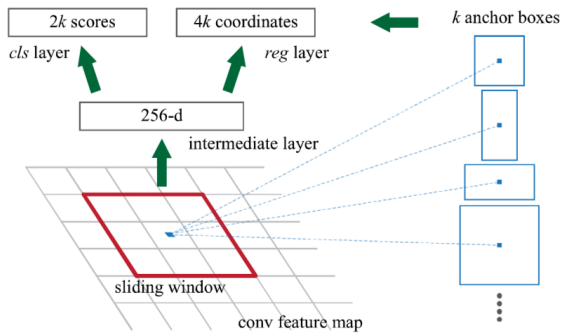

Figure 3. Sliding Window over Anchors (Ren et al., 2015)

A foreground classification (a positive anchor) tells us there is an object within this bounding box. However, the bounding box for this anchor may not be perfectly aligned to the nucleus and is thus improved upon by the regression layer of the RPN (as shown in Figure 5).

2.3. Network Heads

The successfully classified and refined bounding boxes from the RPN are referred to as Regions of Interest (RoIs). These RoIs are now passed on to our network heads which further build on the FPN feature maps. Since both the classifier and bounding box regressor incorporate a fully connected layer in their implementation, we require some form of pooling to achieve a fixed size input. He et al. (2017) proposes RoI Align, an improvement on the RoI pooling (RoI pool) originally used in Fast R-CNN(Girshick, 2015; He et al., 2017). In RoI pool, each RoI is quantised to the spatial granularity of the feature map. Then they are further quantised by sub-dividing into four cells. In instance segmentation, this introduces misalignments between the

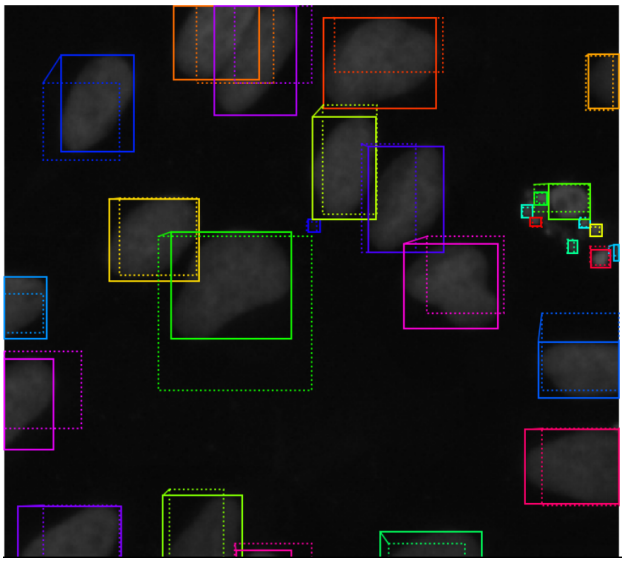

Figure 4. Bounding Box Refinement (dotted lines are bounding boxes before refinement; solid boxes are after refinement by RPN)

RoI and feature map. RoI Align proposes to fix this by sampling each cell at 4 points and calculates their value by bilinear interpolation. The resultant pooled features are of dimension 7×7.

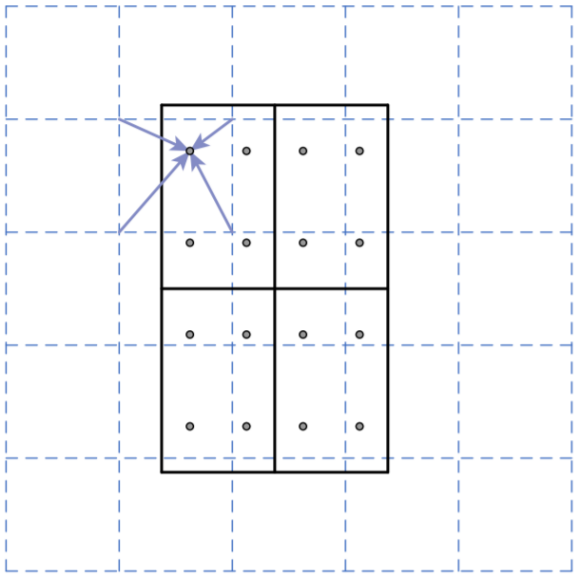

Figure 5. RoI Align (Dashed line - Feature Map, Solid line - RoI and cells, Dots- Sampling points (He et al., 2017)

2.3.1. Classifier and Bounding Box Head

Both the classifier and bounding box head stem from a series of fully connected (FC) layers (Girshick, 2015). These then split as shown in Figure 6. The classifier uses a softmax layer such that it may output class probabilities for a variety of objects. In our implementation we retain the foreground and background classes previously used in the RPN network as there is merely a single class of object

(nucleus). Similarly, the bounding box regressor outputs further refined co-ordinates for the RoI.

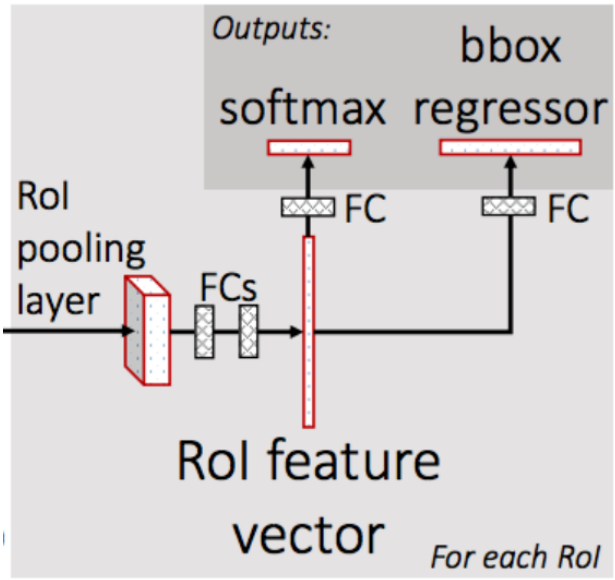

Figure 6. Class and Bounding Box Regression Heads (Girshick, 2015)

2.3.2. Mask Head

Finally we come to the mask head. This is a stack of four convolutional, and ReLU layers followed by a deconvolution and a 1×1 convolution with depth equal to the number of object classes; in our case one.(Nair & Hinton, 2010). The loss function used for the mask head is binary cross-entropy loss:

$$Loss = -(Clog(p(C)) + (1 - C)log(1 - p(C)) \quad (1)$$

where:
C is the true class label,
$p(C)$ is the output classification of the object

This is possible since our mask is independent of the classifier head. In other words, we are only looking at pixel-to-pixel segmentation, irrelevant of the class. Previous applications of Fully Convolutional Networks (FCNs) to segmentation problems have used multinomial cross entropy loss (Long et al., 2015). This has the disadvantage of forcing masks to compete between classes (He et al., 2017). Unfortunately, this design advantage is rendered useless in our application by the fact that we only have one class - nucleus.

2.4. Pre-processing

During our baseline experiments in coursework 3, we showed that the use of *Contrast Limited Adaptive histogram equalisation* (CLAHE) as a pre-processing technique improves the accuracy of the segmentation (Pisano et al., 1998). For this reason, we have automatically extended this technique to the training and inference of our Mask R-CNN model.

2.5. Transfer learning

Transfer learning makes use of the properties of trained weights, by exploiting the fact that learned features from other models may be transferable to the used model(Bengio, 2012). In the case of semantic segmentation, using the weights of a model with the same task makes use of the same explanatory factors. We are making use of transfer learning by using pre-trained weights from the COCO dataset (Lin et al., 2014).

In a perfect world, we would be using pre-training from a biomedical dataset. However, this is not easily accessible with many of these datasets being proprietary. Yet, we chose to pre-train on COCO because we theorise that at the microscopic level we observe features that are similar to those found in the macroscopic level (e.g. circular frisbee). Therefore, we expect this initialisation to be superior to a random one.

2.6. Data augmentation

CNNs are known to be most successful when trained on a large amount of data. Unfortunately our dataset is rather small but data augmentation techniques help to increase the size by creating new images from the available ones. Common techniques are transformations such as flipping, rotating, zooming, cropping or warping the image. Their use has proven to improve accuracies of models (Wang & Perez, 2017; Ronneberger et al., 2015; Cui et al., 2018). Without data augmentation the model may suffer from overfitting (Cui et al., 2018). To increase our dataset size of 670 segmented training images, we rotate and flip the images to make the model rotation invariant. These techniques allow for better generalisation as it makes the model more robust to image variations (Ronneberger et al., 2015). This is particularly useful in nuclei slides since the images are the equivalent of an aerial image where the nucleus will usually rotate in 2D. Moreover, images of nuclei in the test set can come from many different sources so we need to prevent overfitting as much as we can.

2.7. Evaluation

Similar coursework 3, we are using the Intersection over Union (IoU) metric to evaluate the results (Rahman & Wang, 2016; Ronneberger et al., 2015). The equation is described as follows:

$$IoU(Mask, Prediction) = \frac{Mask \cap Prediction}{Mask \cup Prediction} \quad (2)$$

It is important to note that the results of IoU are evaluated on an independent test set for which the ground truth masks are not available to us. External evaluation is done by Kaggle[2].

[2]https://www.kaggle.com/c/data-science-bowl-2018

3. Experiments

As stated previously, Mask RCNN achieved state-of-the-art results on traditional image segmentation task. With these experiments we want to verify that this model can be used for nuclei segmentation and more generally image segmentation for medical images.

Mask RCNN is a very complex model and uses many hyper parameters. For the sake of clarity, we choose to focus on four of them in this report: the non-maximum suppression threshold, the number of RoI and anchor size, and the size of the images. Our choice is motivated by the fact that our dataset is inherently different to the COCO or ImageNet datasets, which have been successfully segmented by the Mask R-CNN model. These datasets contain everyday life objects, persons and animals. Therefore, applying it on the nuclei images requires a change in the hyper parameters such that they suit the dataset better.

3.1. Non-maximum suppression threshold

MOTIVATION

Due to the fact that nuclei may be located in close proximity to each other and not necessarily in a nice horizontal way, it is important to allow bounding boxes that overlap.

DESCRIPTION

In order to allow overlapping bounding boxes, we modified the non-maximum suppression threshold. This hyper parameter controls the maximum IoU that we accept between two RoI before we discard one.

RESULTS

Our first experiment shows that with a threshold of 0.2, the default value in our model's implementation, we end up discarding some overlapping nuclei as we can see in Figure 7. Even though Ren et al. (2015) used 0.7 we choose 0.6 after experimenting with the dataset in data visualisation.

3.2. Number of RoI and Anchor size

MOTIVATION

Another important property of our dataset is that there are potentially hundreds of nuclei on one image, which is higher than the typical number of objects we can find in a photo. Also, the size of a nucleus varies a lot and we can have both instances of a few pixels or masks bigger then 100 pixels.

DESCRIPTION

We increased the number of RoI that we keep from the RPN, as well as the number of instances used for detection and the number of masks for training. It is also important to note that smaller anchors mean potentially more RoI and generally a finer grain.

For the anchor sizes, we experiment with three sets of anchors :

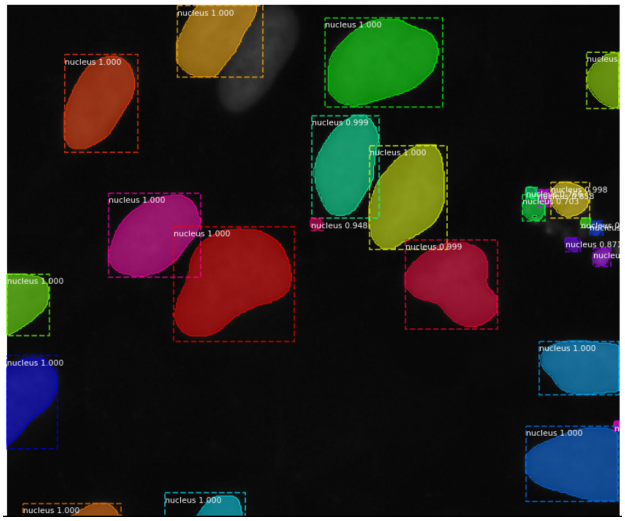

Figure 7. Non detection of a nuclei with a NMS threshold of 0.2 (Top center nucleus not detected, due to NMS shown in 5

- (16, 32, 64, 128, 256)
- (8, 16, 32, 64, 128)
- (4, 8, 16, 32, 64)

We are limited in the number of sizes of anchors we can use by the architecture of the Feature Pyramid Network which contains five layers. We keep the default aspect ratios from our implementation of 0.5, 1 and 2 for the anchors because it matches well with nuclei which generally have a round or elongated shape.

RESULTS

The experiments shown in Table 2 that having smaller anchors did not increase our model's ability to detect nuclei. We discuss this in Section 3.4.

3.3. Image size

MOTIVATION

The nuclei dataset contains images of different sizes that are all resized to the same size to allow the model to process them. With this experiment, we aim to explore whether increasing the image size to the largest images in the dataset helps improve the models performance. The rationale behind this experiment is that by down scaling the image, we may be losing some useful information. Thanks to our pre-processing which improves the contrast of the images, we are able to increase the size of the images without losing too much information in order to enhance the details and help the challenging task of differentiating nuclei that are overlapping or clustered.

Model	Image Resize Scale	NMS	Anchor Scales	IoU Score
UNet	128×128	NA	NA	0.315
	512×512	NA	NA	0.354
Mask R-CNN	512×512	0.2	(8,16, 32, 64, 128)	0.365
	512×512	0.6	(8, 16, 32, 64, 128)	**0.379**
	512×512	0.6	(4, 8, 16, 32, 64)	**0.379**
	1024×1024	0.6	(8, 16, 32, 64, 128)	0.359
	1024×1024	0.6	(16, 32, 64, 128, 256)	0.354

Table 2. Results of our experiments with Mask-RCNN

DESCRIPTION

The Mask R-CNN model resizes each image to the predefined square size. It does so by interpolating one side to the required dimension and if the remaining side is not the same, it will be padded with zeros.

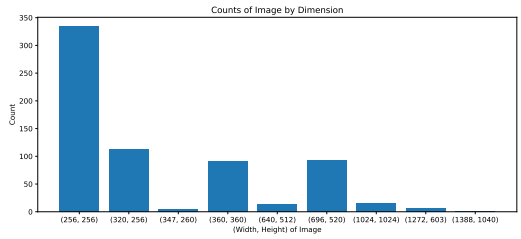

Figure 8. Distribution of image sizes in dataset

In Figure 8, we can see the different sizes of images. By changing the resize settings to 1024 x 1024, we want to see the impact on performance.

RESULTS

By increasing the image size, we could not see an improvement in the IoU score. However, it has a proportional impact on the time required to train the model.

3.4. Interpretation and Discussion

Because nuclei can be located close to each other or even overlapping, we could see that a higher non-suppression threshold is necessary to detect and segment all nuclei.

Due to the relatively small size of nuclei in the images, best performance was reached with anchor sizes of (4, 8, 16, 32, 64) and (8, 16, 32, 64, 128) for image sizes of 512 x 512. The same IoU score was reached for both settings. This suggests that anchor sizes of 4 and 128 are irrelevant for the nuclei detection task at this scale. Results may be improved by choosing anchor sizes that are constant for each layer in the FPN, e.g anchor sizes of (32, 32, 32, 32, 32) or (64, 64, 64, 64, 64) which would allow all the scale invariance to be performed by the FPN. Future experiments need to validate this hypothesis.

It occurs that increasing the image size does not improve the performance of our model. This may be explained by the fact that by increasing the model size, we need to add more padding to smaller images. This may lead to an increased imbalance in our classes at the FPN stage. This has been shown to be detrimental to the performance of convolutional networks (Masko & Hensman, 2015).

Alternatively, it is possible that larger images require more fine tuning of parameters, which we haven't explored in our experiments. For example, it is likely that since we have smaller anchors, many of them will detect the same nuclei and have a high IoU against the ground-truth. It is therefore a sensible thing to do to increase the non-max suppression threshold to decide whether we keep an RoI compared to the ground truth. Since we have a finer grain, we can be more aggressive in the process of discarding RoI to only keep the absolute best.

One other simple explanation is that the edges of our nuclei become blurry when we upscale small images so the bounding boxes regression fails to capture nuclei accurately.

Due to the few amount of large images in the dataset (Figure 8), we can conclude that increasing the image size is not beneficial as it is increasing training time for lower performance. For datasets with a larger amount of large images, this still needs to be validated.

Overall, we saw that training the Mask R-CNN model is challenging due to its vast amount of parameters. Mask R-CNN has 63,733,406 trainable parameters (vs. 1,941,105 trainable parameters in U-Net), which need to be carefully considered for a new task. Although we saw a modest improvement over our similarly scaled U-Net, we expect the model to be improved further given more computing resources. Mask R-CNN provides a framework for more complex instance segmentation at the cost of increased training time and computational cost.

4. Related work

Even though there is no published work on the same dataset, different approaches have been reported when it comes to medical image segmentation.

One of the best know models for medical image segmentation is the U-Net model, which we have implemented as our baseline model (Ronneberger et al., 2015). It has proven to be an effective model when it comes to semantic segmentation. In coursework 3 we could show that with the use of data augmentation we could already see a significant improvement in performance. Their architecture has been widely used and adapted.

Çiçek et al. have shown that by feeding the U-Net model with 2D annotated images of different views of a 3D volume, U-Net was able to provide a 3D segmentation (Çiçek et al., 2016). A similar approach was taken by Milletari et al. who introduced V-Net, an adapted U-Net architecture using 3D convolutional layers to achieve 3D image segmentations (Milletari et al., 2016).

One adaptation of U-Net combined pre-processing and post-processing methods with a nuclei-boundary model, improving the separation of overlapping and touching nuclei. This allowed for a more accurate segmentation of the individual nucleus. Cutting large images into patches, they could segment extra large images which U-Net was unable to process due to limited GPU memory, using patch-wise segmentation. (Cui et al., 2018)

Recently, Drodzal et al. proposed using the U-Net model for pre-processing the images. After the pre-processing, the images are refined and segmented in a following ResNet. By adding skip connections, known from ResNets, Drodzal et al. found an improvement to the U-Net performance. (Drozdzal et al., 2018)

Not just fully convolutional models such as U-Net are used for medical image segmentations tasks. Recently, Recurrent Neural Networks (RNNs) have also been used for this task. Xie et al. adapted clockwork RNNs (Koutnik et al., 2014) by adding a spatial dimension, allowing for a 2D domain. By doing so, they exploit the global semantics of the image. (Xie et al., 2016)

All the above approaches have one thing in common; the sparse availability of annotated data. The use of data augmentation method expands the amount of images at hand by e.g rotating, flipping or warping them. These simple methods allow for a larger training sample. Ronneberger et al.'s state that elastic deformation transformation is the most important method for a limited amount of annotated images (Ronneberger et al., 2015). Unlike Ronneberger et al., Cui et al. showed that rotating images brings the best performance improvement (Cui et al., 2018). However, only by combining all data augmentation methods, the model does not overfit.

Furthermore, the ability to generalise is an important factor for medical image segmentation, as there is a need for accurate predictions for images that vary in quality. Irshad et al. reviewed different data pre-processing techniques, such as Illumination Normalisation, Color Normalisation, Noise Reduction and Image Smoothing with the use of e.g. Gaussian filters and found that their use can improve the segmentation accuracy (Irshad et al., 2014). It was shown that these techniques help improving the generalisation accuracy.

5. Future work and Discussion

We have shown that our Mask R-CNN implementation exceeds the performance of our U-Net baseline model. However, the results suggest there is still a ways to go before near perfect generalisation is achieved. As the literature shows, combining our current methods with elastic deformation transformations may improve our generalisation.

From Section 4 we can see that the majority of approaches makes use of fully convolution network architectures such as U-Net. For coursework 3, we have shown that U-Net is an effective segmentation tool, however, needs improvements to better generalise to varying image qualities and staining techniques(Aquilina et al., 2018). By combining it with more pre-processing, data augmentation, fine-tuning and post-processing techniques, the literature has shown its potential in medical image segmentation and gives reason to assume that it can be improved even further.

The advantage of a U-Net-like architecture is its computational efficiency and lightweight implementation. Training time for our U-Net architecture was substantially shorter with approximately 45 minutes for images resized to 128x128 compared to 8-10 hours for Mask R-CNN.

For a nuclei image segmentation task, using an adaptation of U-Net such as the one presented by Drodzal et al. (Drozdzal et al., 2018) may be more feasible than using Mask R-CNN. Future work should focus on comparing an adapted and optimised U-Net architecture-based model with the Mask R-CNN implementation. Using Mask R-CNN for the given task only makes sense if it provides a substantially better performance, as training and optimisation is expensive.

As discussed in Section 2, Mask R-CNN has the ability to classify and segment multiple object classes. For this dataset, we are dealing with only 1 object class: 'nuclei'. However, Mask RCNN is capable of distinguishing between a lot more classes. It would be interesting to apply Mask RCNN on a dataset that does not only have the masks for nuclei but also classifies a state of the nuclei e.g. whether it is cancerous and the type of cancer.

6. Conclusion

In this paper we have applied the Mask RCNN for nuclei segmentation. Though, a lot more computationally expensive, Mask R-CNN outperforms our U-Net baseline model on an independent test set. We can therefore conclude, that Mask R-CNNs potential can indeed be translated to the field of medical image segmentation. Due to its high complexity, it can be assumed that further improvements can be made by hyperparameter optimisations. Furthermore, the literature suggests that further use of data augmentation and both pre and post processing can improve its performance.

Future work should validate its superiority compared to improved and adapted U-Net architectures, as those are easier to train and more computationally efficient for this task.

With our work, we have shown that Mask R-CNN is an effective model for nuclei segmentation. This opens up new possibilities for medical research as it may also be capable

of differentiating between different kind of nuclei.

References

Aquilina, Andre, Glombek, Marieke, and Ramirez, Anthony. Mlp coursework 3. 2018.

Bengio, Yoshua. Deep learning of representations for unsupervised and transfer learning. In *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, pp. 17–36, 2012.

Çiçek, Özgün, Abdulkadir, Ahmed, Lienkamp, Soeren S, Brox, Thomas, and Ronneberger, Olaf. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 424–432. Springer, 2016.

Cui, Yuxin, Zhang, Guiying, Liu, Zhonghao, Xiong, Zheng, and Hu, Jianjun. A deep learning algorithm for one-step contour aware nuclei segmentation of histopathological images. *arXiv preprint arXiv:1803.02786*, 2018.

Drozdzal, Michal, Chartrand, Gabriel, Vorontsov, Eugene, Shakeri, Mahsa, Di Jorio, Lisa, Tang, An, Romero, Adriana, Bengio, Yoshua, Pal, Chris, and Kadoury, Samuel. Learning normalized inputs for iterative estimation in medical image segmentation. *Medical image analysis*, 44:1–13, 2018.

Girshick, Ross. Fast r-cnn. *arXiv preprint arXiv:1504.08083*, 2015.

Girshick, Ross, Donahue, Jeff, Darrell, Trevor, and Malik, Jitendra. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587, 2014.

He, Kaiming, Zhang, Xiangyu, Ren, Shaoqing, and Sun, Jian. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

He, Kaiming, Gkioxari, Georgia, Dollár, Piotr, and Girshick, Ross. Mask r-cnn. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pp. 2980–2988. IEEE, 2017.

Irshad, Humayun, Veillard, Antoine, Roux, Ludovic, and Racoceanu, Daniel. Methods for nuclei detection, segmentation, and classification in digital histopathology: a review - current status and future potential. *IEEE reviews in biomedical engineering*, 7:97–114, 2014.

Koutnik, Jan, Greff, Klaus, Gomez, Faustino, and Schmidhuber, Juergen. A clockwork rnn. *arXiv preprint arXiv:1402.3511*, 2014.

Lin, Tsung-Yi, Maire, Michael, Belongie, Serge, Hays, James, Perona, Pietro, Ramanan, Deva, Dollár, Piotr, and Zitnick, C Lawrence. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Lin, Tsung-Yi, Dollár, Piotr, Girshick, Ross, He, Kaiming, Hariharan, Bharath, and Belongie, Serge. Feature pyramid networks for object detection. In *CVPR*, volume 1, pp. 4, 2017.

Long, Jonathan, Shelhamer, Evan, and Darrell, Trevor. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

Masko, David and Hensman, Paulina. The impact of imbalanced training data for convolutional neural networks, 2015.

Milletari, Fausto, Navab, Nassir, and Ahmadi, Seyed-Ahmad. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *3D Vision (3DV), 2016 Fourth International Conference on*, pp. 565–571. IEEE, 2016.

Nair, Vinod and Hinton, Geoffrey E. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pp. 807–814, 2010.

Pisano, Etta D, Zong, Shuquan, Hemminger, Bradley M, DeLuca, Marla, Johnston, R Eugene, Muller, Keith, Braeuning, M Patricia, and Pizer, Stephen M. Contrast limited adaptive histogram equalization image processing to improve the detection of simulated spiculations in dense mammograms. *Journal of Digital imaging*, 11(4):193, 1998.

Rahman, Md Atiqur and Wang, Yang. Optimizing intersection-over-union in deep neural networks for image segmentation. In *International Symposium on Visual Computing*, pp. 234–244. Springer, 2016.

Ren, Shaoqing, He, Kaiming, Girshick, Ross, and Sun, Jian. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pp. 91–99, 2015.

Ronneberger, Olaf, Fischer, Philipp, and Brox, Thomas. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Wang, Jason and Perez, Luis. The effectiveness of data augmentation in image classification using deep learning. Technical report, Technical report, 2017.

Xie, Saining, Girshick, Ross, Dollár, Piotr, Tu, Zhuowen, and He, Kaiming. Aggregated residual transformations for deep neural networks. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pp. 5987–5995. IEEE, 2017.

Xie, Yuanpu, Zhang, Zizhao, Sapkota, Manish, and Yang, Lin. Spatial clockwork recurrent neural network for muscle perimysium segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 185–193. Springer, 2016.